Analysis of French emergency housing requests data

Alexis Bogroff*

May 2, 2020

1 Introduction

This report is an enhanced and concise aggregation of the comments that can be found along the code, while presenting explanations on the decisions made. It also gives a quick overview of the project.

2 Project architecture

The code is divded into:

- data_exploration.ipyb: the main notebook file
- cobratools.py: a module containing the classes and functions

The code files are attached in the mail of submission, but they can still be found in the following Github repository (not the data):
github.com/AlexisBogroff/Contests/tree/master/Predictions/Emergency_housing

The code is mainly based on the open-source python libraries:

- numpy
- pandas
- Pytorch

3 Data

3.1 Principal components

Once applied the pre-processing describe later, the following extract of the correlation matrix from the train set present the most impactful variables with an absolute correlation above 4% with the target variable:

*mail: alexis.bogroff@gmail.com

46.8	district_grant_ratio
44.44	town_grant_ratio
35.85	housing_situation_label_cat_hotel paid by the emergency centre
25.46	housing_situation_label_cat_emergency structure
11.87	group_creation_month
8.85	group_composition_label_cat_single mother with child(ren)
4.96	housing_situation_label_cat_hotel paid by the regional administration
4.64	victim_of_violence
-4.02	group_composition_label_cat_group of adults
-4.38	answer_creation_month
-4.43	request_creation_month
-4.62	group_composition_label_cat_couple without whildren
-4.64	victim_of_violence_type_cat_no violence
-4.84	housing_situation_label_cat_mobile or makeshift shelter
-6.09	group_composition_label_cat_man alone
-8.58	housing_situation_label_cat_accomodation by a third party
-8.64	group_creation_year
-37.75	housing_situation_label_cat_street

Based on this table, any character of emergency, weakness and loneliness has a high positive impact on the target, while signs of supporting entourage and wealth impacts it negatively. Data is coherent with the goal of the institution.

3.2 Pre-processing

Methodology:

- Clean-up request dataframe
- Feature engineer (partly using individuals dataset)

3.2.1 Join data sets

Since there are multiple requests by individuals and multiple individuals by request, the straightfoward approach would be to create columns for each individual' informations. This way, no information would be lost, but the curse of dimensionality is very near and the number of samples might be too low to extract useful information.

The chosen approach is rather to only keep the request dataset' columns, and feature engineer additional columns based on the individuals data, eg.: number of past requests made by the same individual, number nights granted in past requests of the same individual(s)/group, gender diversity of the group, etc.

However, for analytics purpose, a dataframe with all the data is also created.

3.2.2 NaNs imputation

Methodology:

1. inspect NaNs on train set
2. if pattern detected, apply modifications on train and test sets

group-composition-id

- what to do: drop group-composition-label
- why: the data seems to derive group-composition-id from group-composition-label, both would then necessary be redondant.

child-to-come

- what to do: impute child-to-come from the pregnancy in the individuals of the request
- why: there are 145947 NaNs for child-to-come on the train set (in request), and only 14 NaNs for pregnancy in train set (in individuals). The former can thus be derived from the latter.

housing-situation-label

- what to do:
  1. impute housing-situation-label NaNs as 'street'
  2. drop housing-situation-id
  3. one-hot encode housing-situation-label
- why:
  1. 90% (21,185) of missing housing-situation-label are housing-situation-2-label "on the street"
  2. housing-situation-id is redondant with housing-situation-label
  3. it must be transform to a variable interpreted as categorical by the model, since if there is a logic in the numerical values of each class, it is not linear.

long-term-housing-request

- what to do: drop feature
- why: it seems to have no direct impact on target

town

- what to do: attribute the most probable town based on request district
- why: it might be highly probable to ask for emergency housing where the individual live.

victim-of-violence-type

- what to do:
  1. Set a specific value to NaNs where victim-of-violence is 'f', which will later be transformed into a boolean
  2. Set another specific value to NaNs where victim-of-violence is 't'
- why: is NaN if victim-of-violence is 'f', and the grand majority of victim-of-violence-type NaNs comes from the absence of violence.

child-situation

- what to do: replace child-situation by 10 when victim-of-violence-type 'child' Idem for 'family'

- why: -1 (NaNs) for child-situation are mainly mapped with 10 when victim-of-violence-type 'child'

Remaining test set NaNs

- what to do: impute with the first sample found in train set (yes it is bad i
- why: NaNs from test set that have no equivalent in train set can't be studied upfront
- improve: allocating the average value from the same group of requests for the given variable

3.2.3 Outliers

Gender

- what to do: set to female when individual is pregnant
- why: only females are possibly pregnant

answer creation date

- what to do: drop the whole feature
- why: the variable is not available at prediction time

3.2.4 Dropped features

- Numerical columns dropped, mostly because it would require time to build larger categories based on there numeric values.
  - district (replaced by district-grant-ratio)
  - town (replace by town-grant-ratio)
  - group-id
  - group-main-requester-id
  - request-backoffice-creator-id
  - social-situation-id

3.2.5 Delete samples

It has not been applied, but to build a ready-for-production system, it should probably ignore old data, as the emergency housing centers evolve along with their selection criteria and capacity.

- Train/test split being done randomly (historically), it is important for this competition to train the model on the whole train set (don't remove old samples).
- Otherwise, delete samples with group-creation-date ¡ 2015, since it is very unlikely that current demands are treated like +5 years ago (social services evolve)

3.3 Feature engineering

Transform current features:

- Transform dates into linear numerical features (year, month)
- Transform categorical features (with less than 30 classes), into one-hot-encodings

Create new features:

- district-grant-ratio and town-grant-ratio:
  - into numerical features linearly separable
  - obs: district-grant-ratio: has a large impact, with districts granting more nights than there are requests and some refusing way more often
  - hyp: sort of district emergency housing capacity measurement against the emergency housing demand

4 Model: construction, training and evaluation decisions

4.1 Architecture and parameters

The retained model is a neural network composed of two hidden layers. It uses Rectified Linear Unit (ReLu) as internal activation functions, and passes its output logits directly to the weighted cross entropy criterion. A softmax is applied to obtain interpretable predictions. A detail information on the final hyper-parameters can be found directly in the section 'Predict' of the notebook.

4.2 Evaluation

I understand the choice of the weighted log-loss criterion as favoring the robustness and security of a model. It favors a model that is not accurate, to a model that yields wrong predictions with strong confidence.
It is however more difficult to reach a good accuracy by setting weights varying largely, and the behavior of the current model decreases accuracy in close relation to the reduction in loss.

5 Bias, interpretability

The choice of a standard neural network is not optimal, and further methods to increase its interpretability should be used. Also, ensemble methods are recognized for their robustness and efficiency to a large variety of problems, hence futher developments would require their implementation. In regard with the split of train set into two parts to produce an evaluation set, the robustness of the predictions is increased, and reduce the out-of-sample variance.

6 Further improvements

Analysis

- explore requester-type data along with group-main-requester-id. If the agent is an urgentist, used to bring individuals to the service, its groups might have higher granted rates than random individuals coming on their own.

- explore request-backoffice-creator-id data. It could impact the result since each people has its own biases (as for the predictions of court decisions). However it is named 'backoffice', and could imply that the person is not in position to impact the decision.

Pre-processing:

- Imputations:
  - Build more robust, generalisable imputations
  - Automate NaNs imputation for future test samples
  - Reconstruct some NaNs by training models to predict the missing feature
  - better impute child-situation (only a minority have been imputed properly)
  - Impute the 14 pregnancy NaNs from child-to-come
  - housing-situation-label: derive more sub-groups, and impute with its most often matched value
- Feature engineering, create new features:
  - town-capacity-remaining: this can be computed on the month or the year. It could provide a guess of the number of nights that can be granted at time of request (using past request).
  - distance between town and district
  - transform town and district to regions
  - number of individuals in the group
  - number of past requests by individuals forming the group of the request
  - same feature for granted requests only
  - hot-cold season

Code:

- Implementation to enable GPU
- Refactor imputation of child-to-come NaNs (takes +3 min)

Model:

- prediction with confidence intervals
- ensemble methods
- methods to increase the interpretability of the neural network

Approach:

- Add domain knowledge into the mix